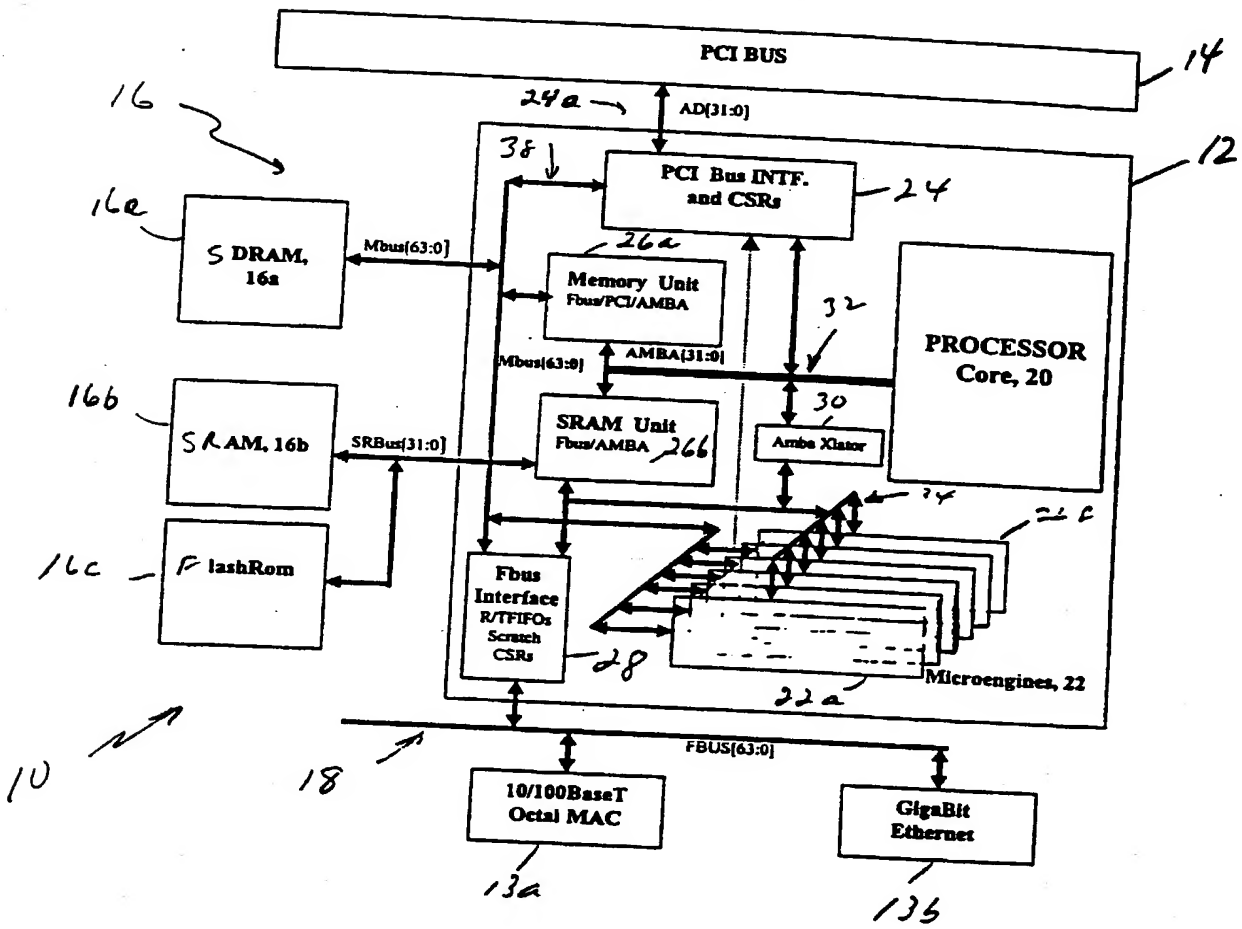

PCI BUS
14
16
24a
AD[31:0]
38
12
PCI Bus INTF. and CSRs
24
16a
SDRAM, 16a
Mbus[63:0]
26a
Memory Unit Fbus/PCI/AMBA
32
PROCESSOR Core, 20
Mbus[63:0]
AMBA[31:0]
16b
SRAM, 16b
SRBus[31:0]
SRAM Unit Fbus/AMBA
26b
30
Amba Xlator
34
22c
16c
FlashRom
Fbus Interface R/TFIFOs Scratch CSRs
28
Microengines, 22
22a
10
18
FBUS[63:0]
10/100BaseT Octal MAC
GigaBit Ethernet
13a
13b

FIG. 1

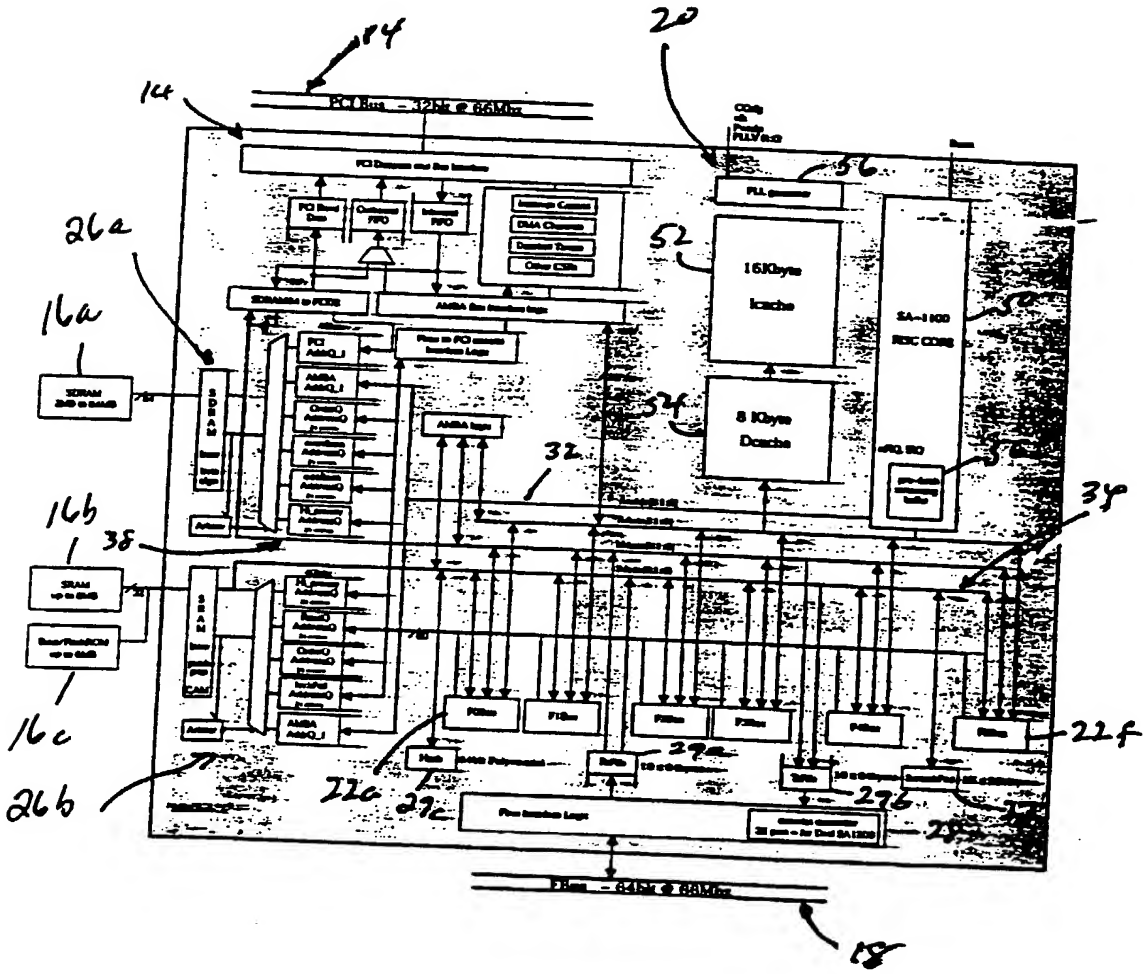

12
20
14
PCI Bus – 32bit @ 66Mhz
56
26a
52
16Kbyte Icache
16a
54
8 Kbyte Dcache
32
34
16b
38
16c
26b
22a
22c
22f
27a
27b
18

FIG. 2

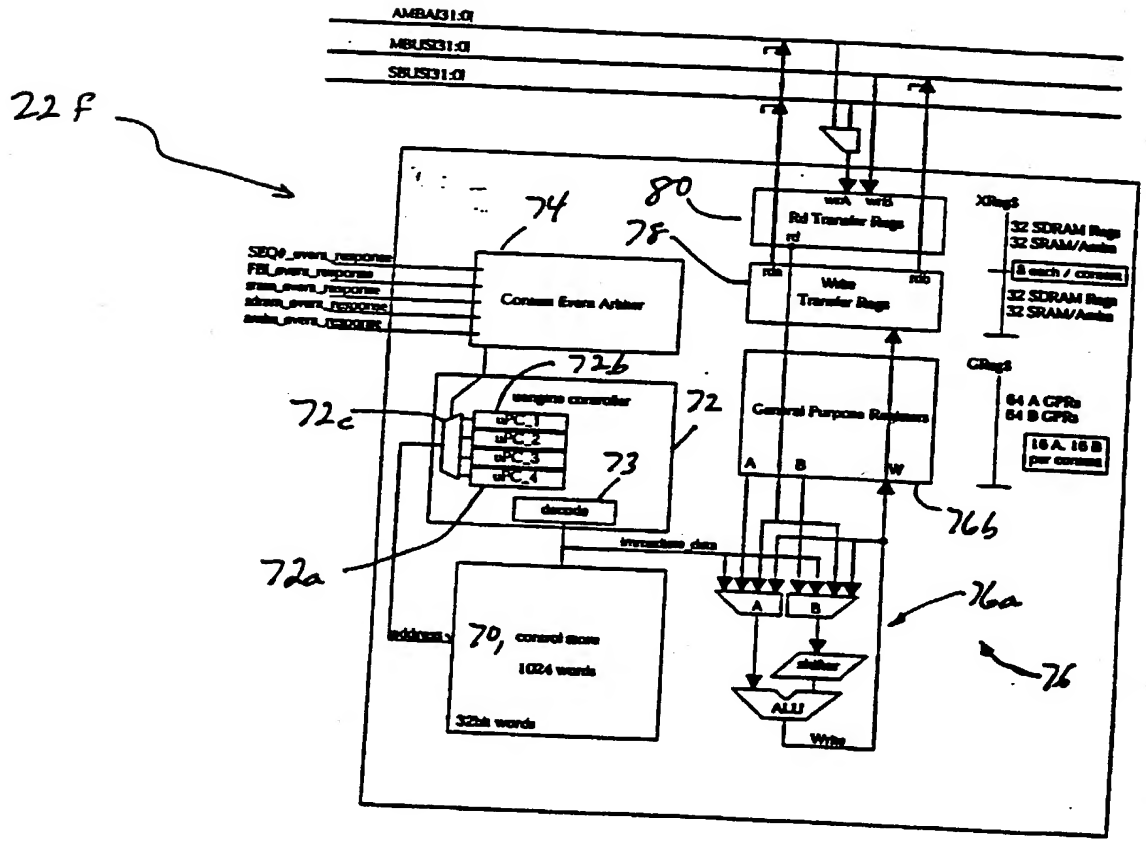

22f
AMBA[31:0]
MBUS[31:0]
SBUS[31:0]
74
80
78
Rd Transfer Regs
Write Transfer Regs
Common Event Arbiter
72b
72c
72
72a
73
uPC_1
uPC_2
uPC_3
uPC_4
decode
General Purpose Registers
A
B
W
76b
76a
76
ALU
shifter
70, control store
1024 words
32bit words
32 SDRAM Regs
32 SRAM/Addr
64 A GPRs
64 B GPRs
16 A, 16 B per context

FIG. 3

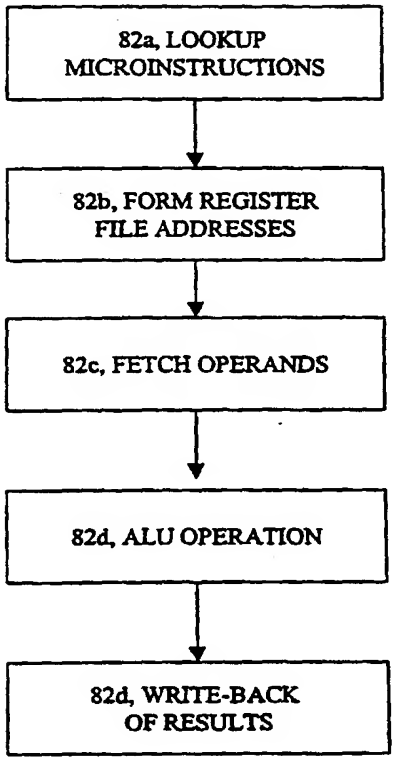
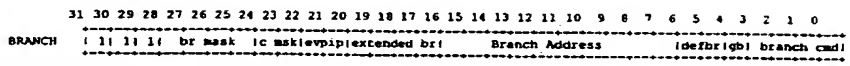

82a, LOOKUP MICROINSTRUCTIONS
82b, FORM REGISTER FILE ADDRESSES
82c, FETCH OPERANDS
82d, ALU OPERATION
82d, WRITE-BACK OF RESULTS

FIG. 4

branch instruction:



defbr: A value of 0, 1 or 2 may be specified. If non-zero, the value indicates that the following 1 or 2 microwords will be allowed to execute before the branch operation takes place.

gb: If set, guess that the branch path will be taken, thus prefetch the branch microword address. Otherwise prefetch the non-branch path. This field is only allowed to be set when defbr=0 or defbr=1.

branch address: branch address conditionally or unconditionally selected.

br_mask: Is decoded to the following options:

1) unconditional branch
2) branch when ALU<31>=1 (<0)
3) branch when ALU<31>=0 (>=0)
4) branch when ALU<31>=1 OR ALU<31:0>=0 (<=0)
5) branch when ALU<31>=0 AND ALU<31:0>!=0 (>0)
6) branch when ALU<31:0>=0 (=0)
7) branch when ALU<31:0>=1 (!=0)
8) branch when specified context mask = current context
9) branch when specified context mask != current context
10) branch on carry-out set
11) branch on carry-out clear
15) look at extended branch field to further decode branch type

extended_br: branches on various context-swapping signals or other signals.

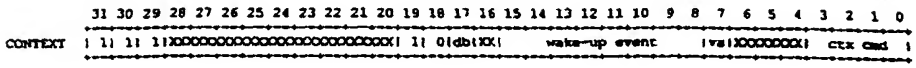
evpip: indicates pipe stage that this branch should be evaluated in

c msk: specifies a context number with which to conditionally branch on.

branch cmd: further specifies the type of branch, e.g., looks at condition codes or some other branch criteria.

FIG. 5A

6/13



Context Descriptors:

1) Wake-up Events

0 = kill	8 = FBI	128 = SEQ_NUM_LSB
1 = voluntary	16 = INTER_THREAD	
2 = SRAM	32 = PCI_DMA_1	
4 = SDRAM	64 = PCI_DMA_2	

2) db -> branch defer amount
3) va -> value of sequence number

FIG. 5B

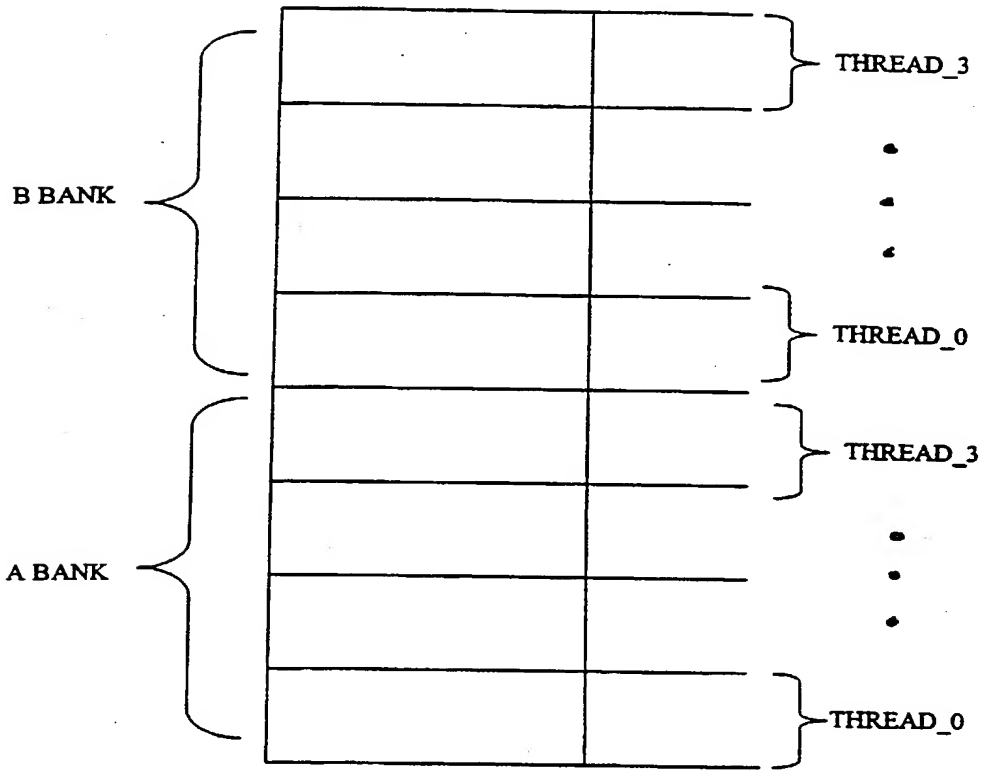


FIG. 6

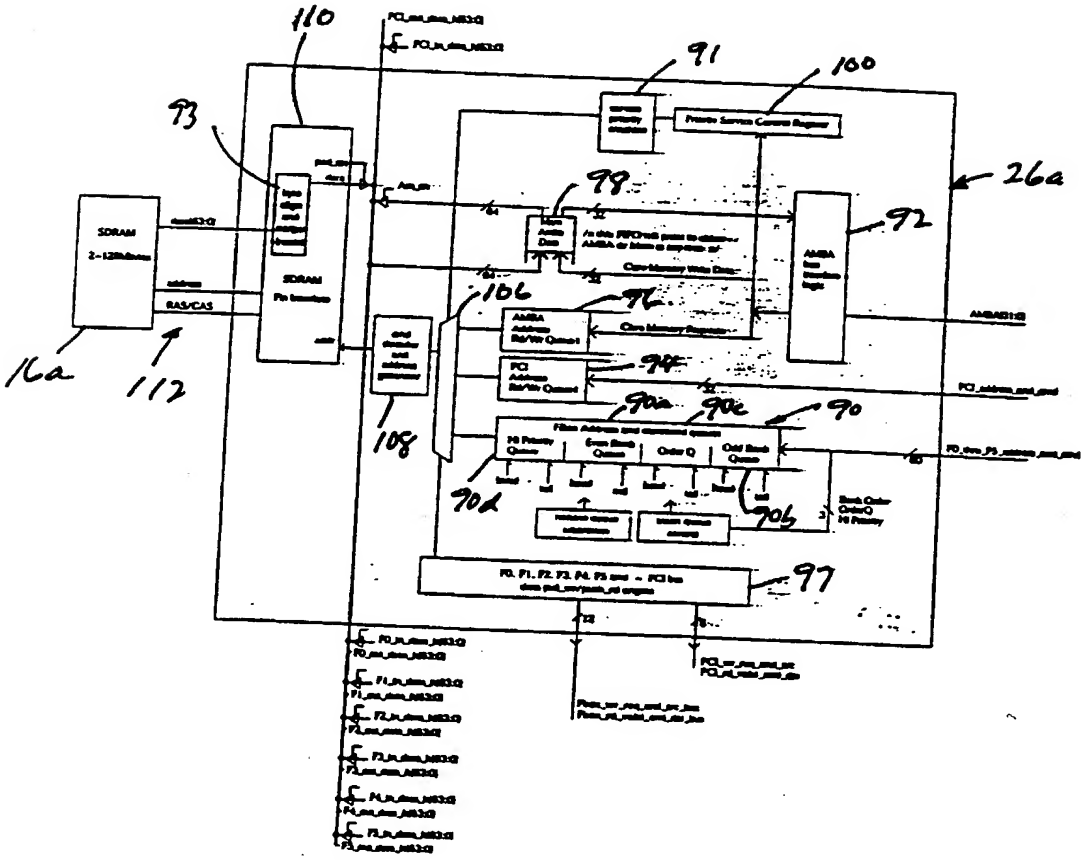

110
91
100
93
99
26a
92
106
96
94
90a
90c
90
108
90d
90b
97
SDRAM
RAS/CAS
16a
112

FIG. 7

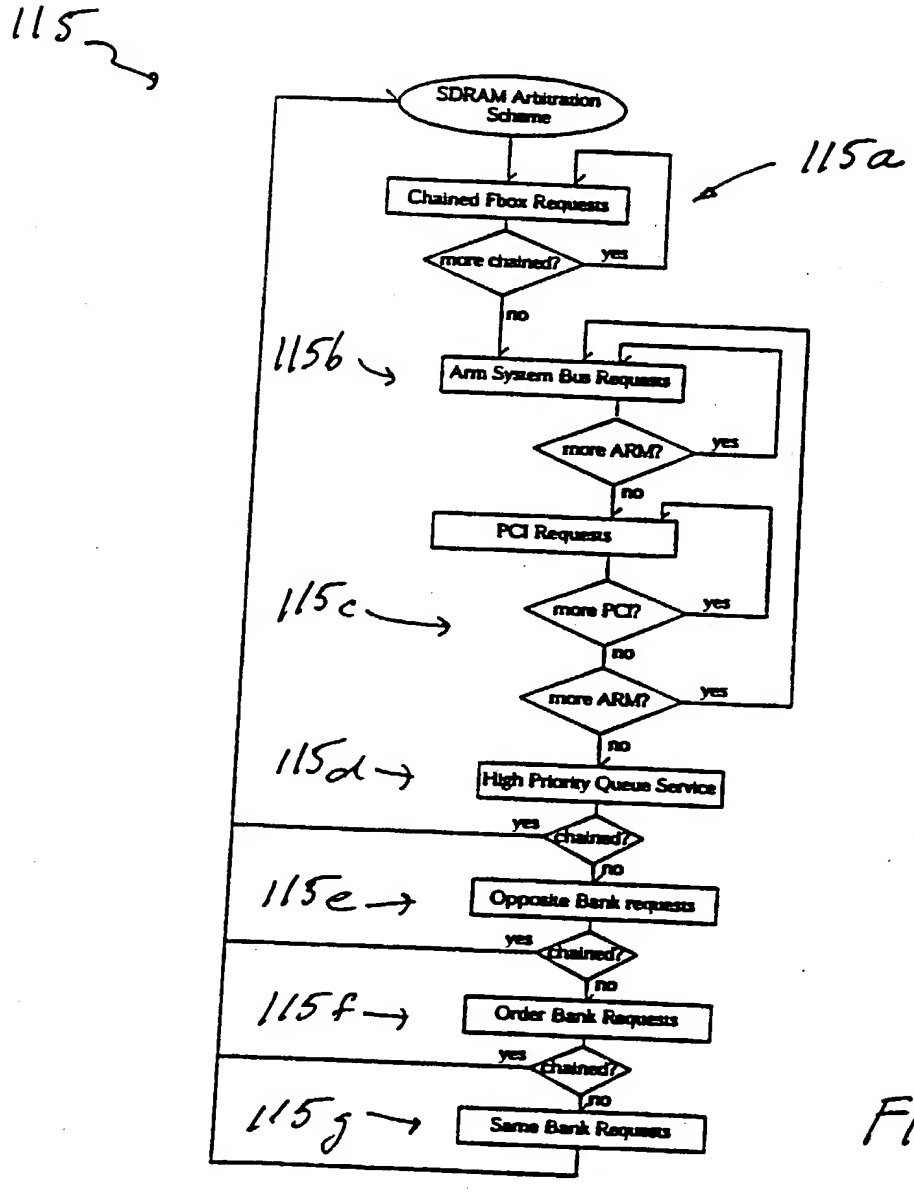

115
SDRAM Arbitration Scheme
115a
Chained Fbox Requests
more chained?
yes
no
115b
Arm System Bus Requests
more ARM?
yes
no
PCI Requests
115c
more PCI?
yes
no
more ARM?
yes
no
115d
High Priority Queue Service
yes
Chained?
no
115e
Opposite Bank requests
yes
Chained?
no
115f
Order Bank Requests
yes
Chained?
no
115g
Same Bank Requests

FIG. 7A

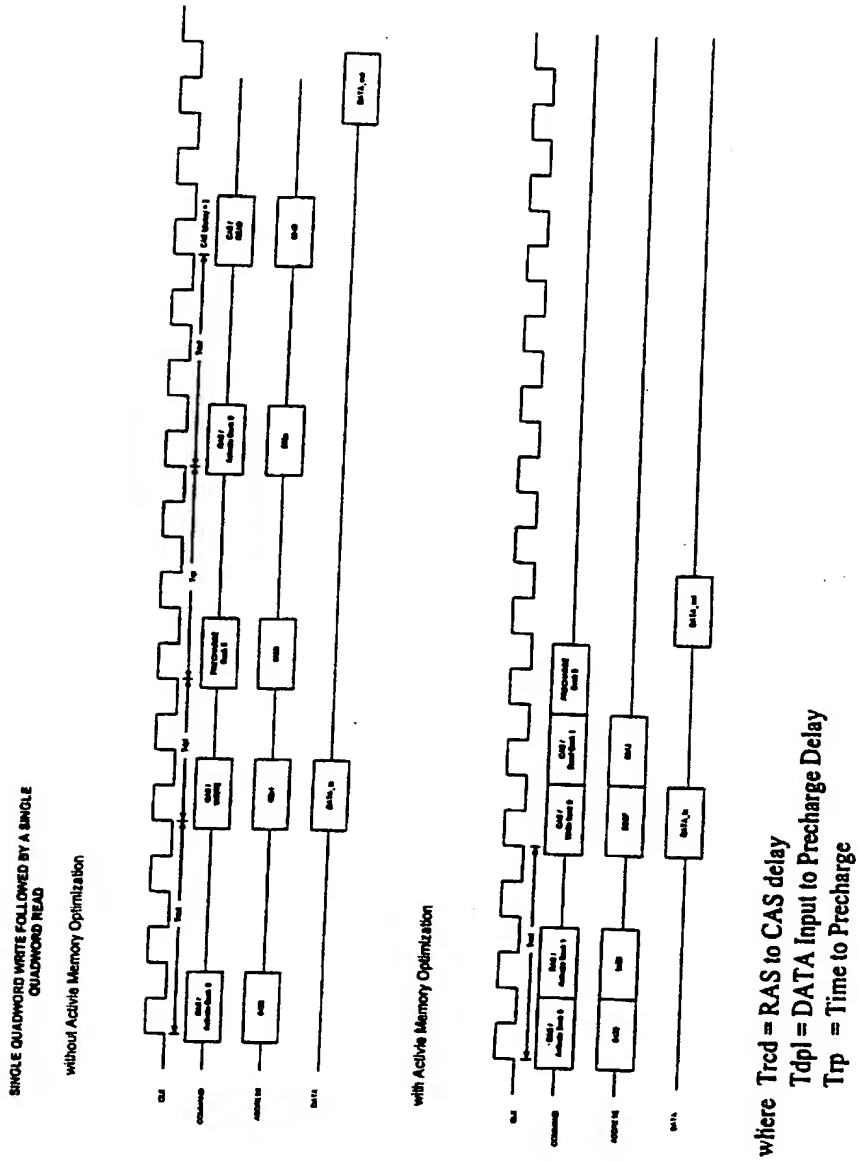

SINGLE QUADWORD WRITE FOLLOWED BY A SINGLE QUADWORD READ
without Activie Memory Optimization
with Activie Memory Optimization
where Trcd = RAS to CAS delay
Tdpl = DATA Input to Precharge Delay
Trp = Time to Precharge

FIG. 7B

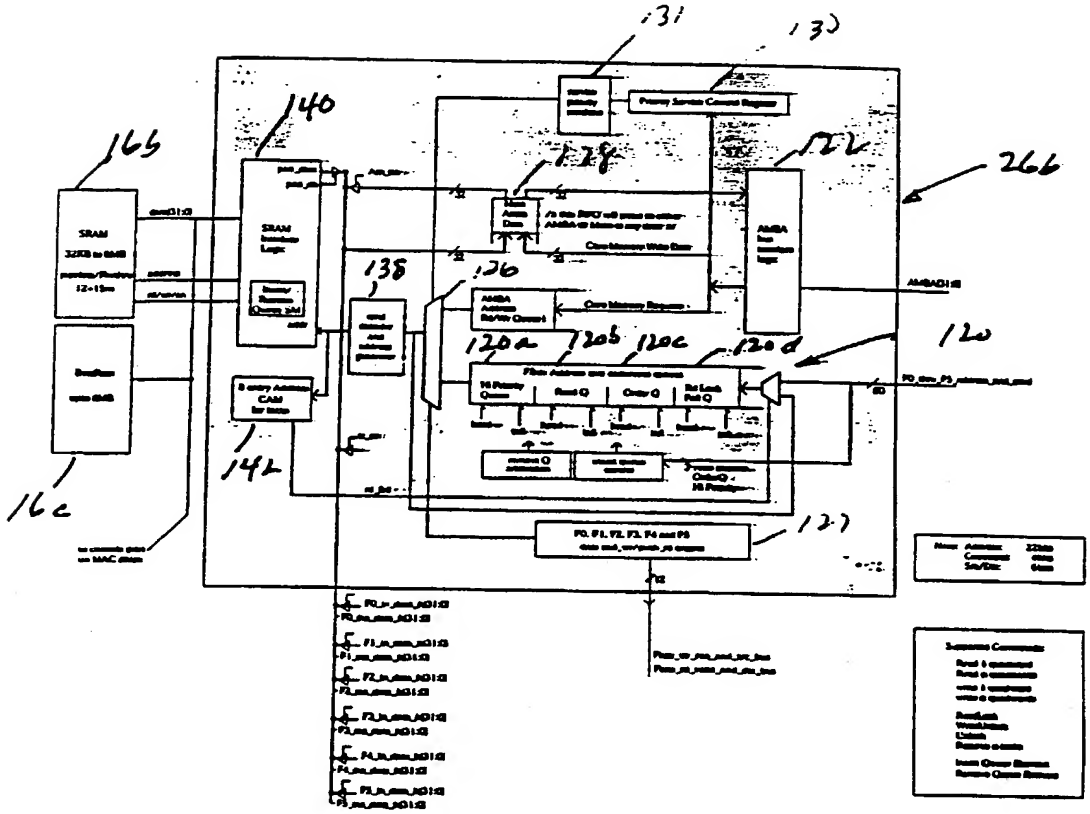


FIG. 8

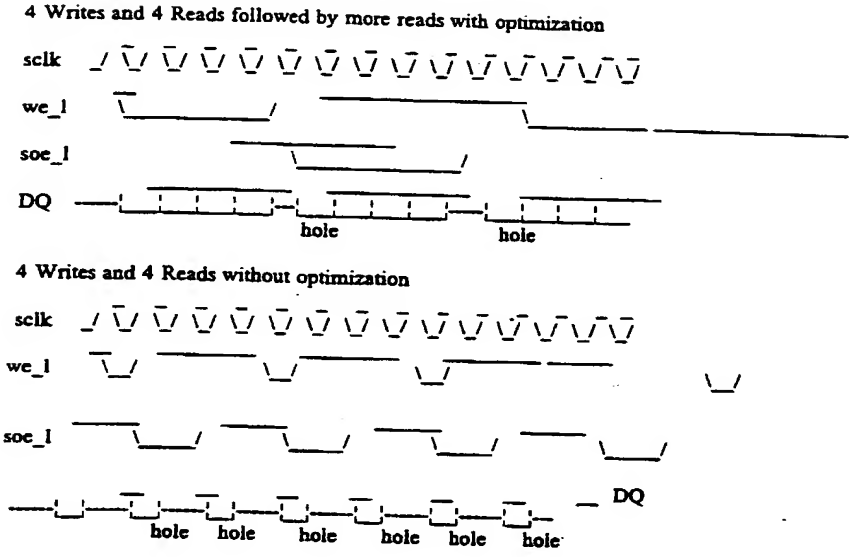

4 Writes and 4 Reads followed by more reads with optimization
sclk
we_l
soe_l
DQ
hole
hole
4 Writes and 4 Reads without optimization
sclk
we_l
soe_l
DQ
hole
hole
hole
hole
hole
hole
10 cycles vs. 14.

FIG. 8A

13/13

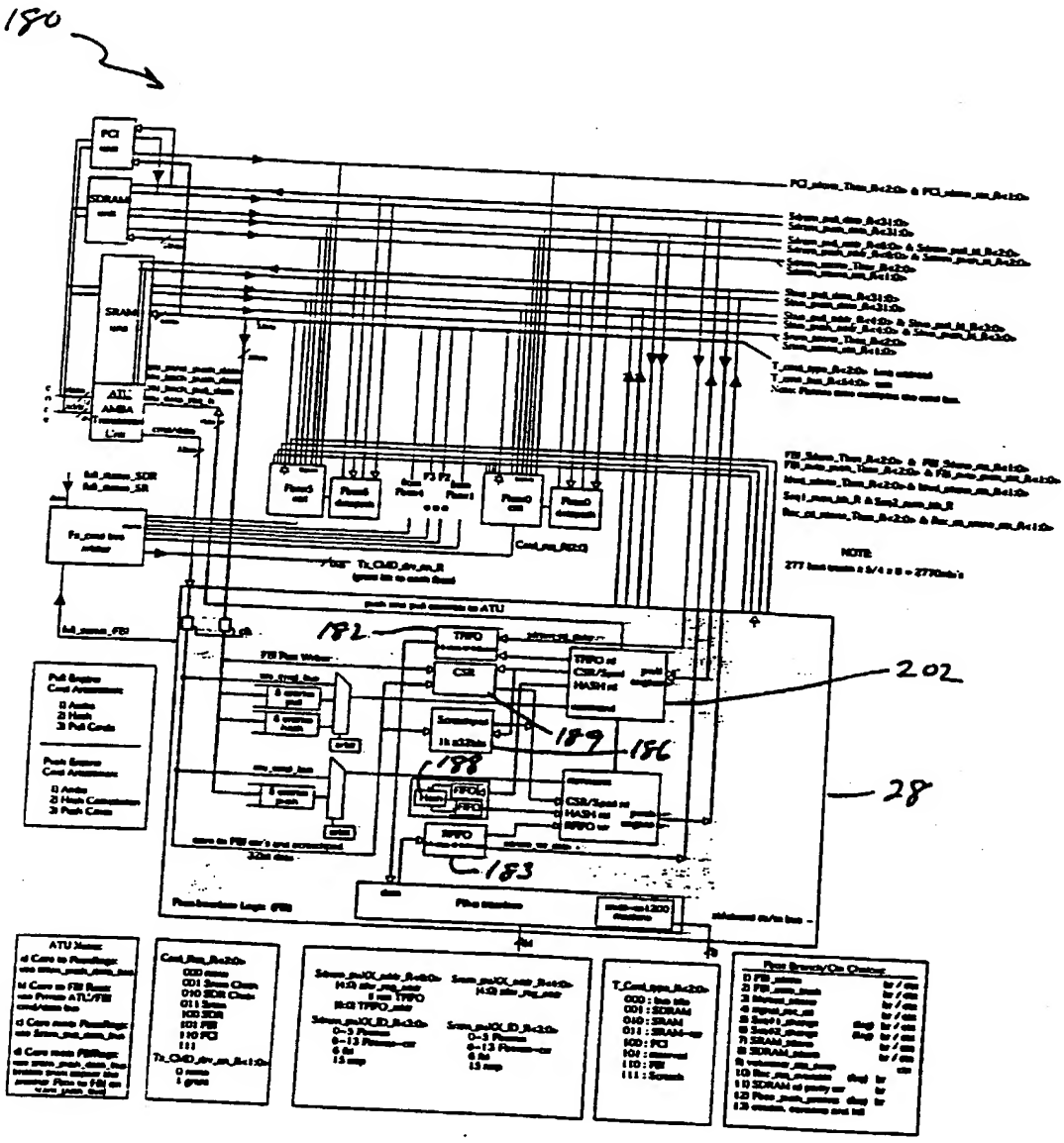


FIG. 9